

**Performance analysis  
Paraver & Dimemas**

Jesús Labarta, Judit Gimenez  
Jordi Caubet, Francesc Escale  
CEPBA-UPC

Technology Transfer      Research |      Training      Mobility of Researchers  
User Support      Education |      HPC Facilities      Parallel Expertise

## Tutorial agenda

**Monday July 22<sup>nd</sup>**

10:00 - 11:00 : Paraver Overview  
11:00 - 12:00 : Dimemas Overview

13:30 - 15:00 : Paraver and Dimemas Demos  
15:00 - 17:00 : Hands on session (I)

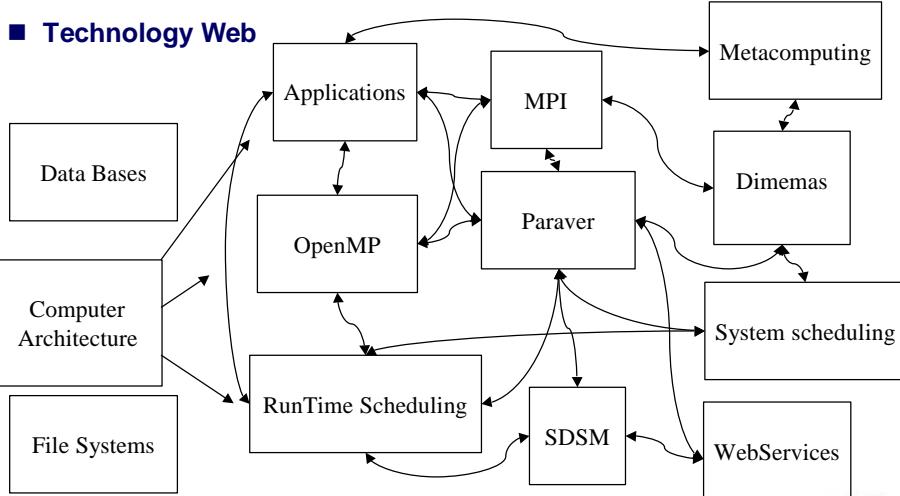
**Tuesday July 23<sup>rd</sup>**

09:00 - 10:00 : Paraver Advanced  
10:00 - 12:00 : Hands on session (II)

Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002



## CEPBA R&D Philosophy



Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002

## Why performance tools?

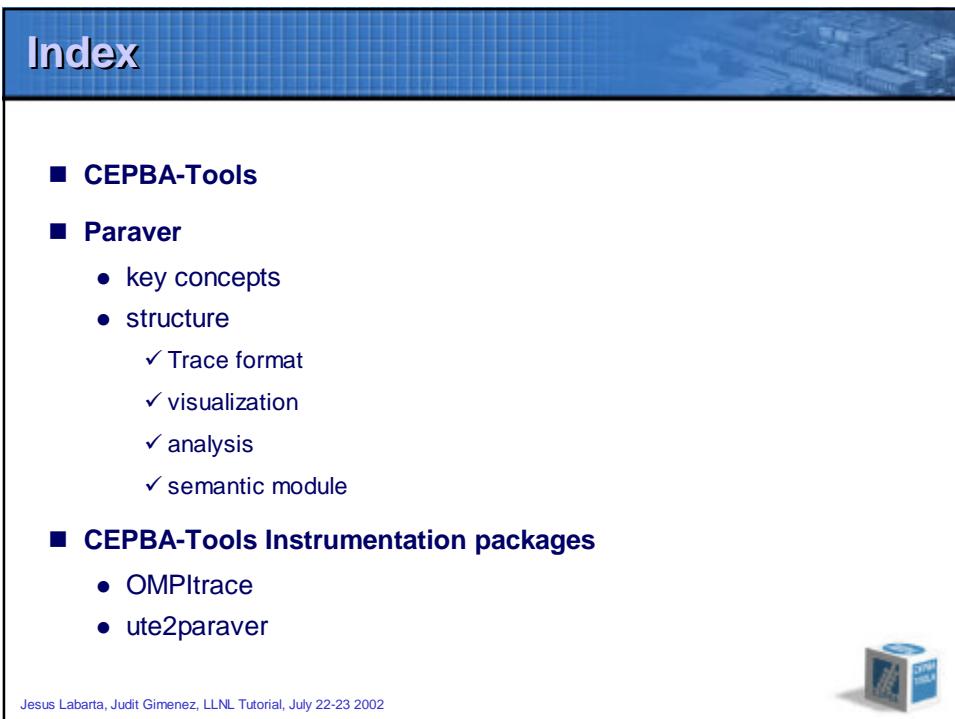
Strategic

Seeing is believing,  
measuring is better

Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002



The slide features a background image of a city skyline at night. In the top left corner, there is a small blue icon of a computer monitor with three vertical bars. Overlaid on the image is the title "Paraver Overview" in a large, dark blue serif font. Below the title, the names "Jesús Labarta, Judit Giménez, Jordi Caubet, Francesc Escale" are listed in a smaller, dark blue sans-serif font. At the bottom center, the text "CEPBA-UPC" is displayed in a dark blue sans-serif font. Along the bottom edge of the slide, there is a horizontal menu bar with several items: "Technology Transfer", "Research", "Training", and "Mobility of Researchers" are grouped together; "User Support", "Education", "HPC Facilities", and "Parallel Expertise" are also grouped together.



The slide has a blue header bar with the word "Index" in white. The main content area contains a list of topics under the heading "■".

- CEPBA-Tools
- Paraver
  - key concepts
  - structure
    - ✓ Trace format
    - ✓ visualization
    - ✓ analysis
    - ✓ semantic module
- CEPBA-Tools Instrumentation packages
  - OMPtrace
  - ute2paraver

At the bottom left, the text "Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002" is visible. On the right side, there is a small blue cube icon with the text "CEPBA-UPC" on it.

## CEPBA-Tools Challenge

**What can we say  
about an unknown application/system  
without looking at the source code  
in short time**

Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002



## Thesis

**"A single instrumented run  
captures a lot of information  
that is essentially thrown away  
in current parallel programming practice"**

Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002



## CEPBA-Tools

### ■ Research on tools and methodology to

- shorten the program tuning cycle
- increase understanding of programs and systems
  - ✓ OpenMP compiler and run time
  - ✓ Scheduling of multiprogrammed parallel workloads

### ■ Core tools

- Paraver
- Dimemas

... available through  
<http://www.cepba.upc.es/tools>

Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002



## Paraver

(1992- )

### ■ Performance visualization tool

- Off line analysis of traces

### ■ Quantitative, comparative

### ■ Powerful: Flexibility !!!

- Performance analysis = search on a huge and fuzzy space
- One would better
  - ✓ Be equipped with flexible tools ...
    - No semantics in the tool
  - ✓ ...supporting quantitative analysis ...
  - ✓ and be ready for surprises

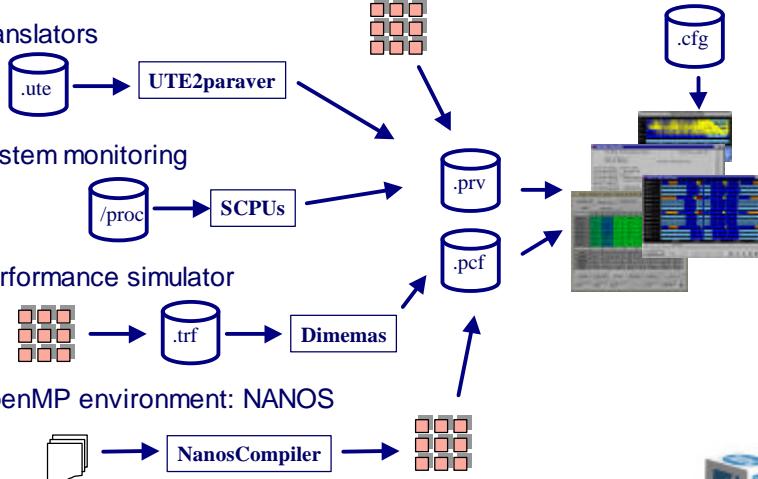
Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002



## CEPBA-Tools

- Tracing tools: OMPtrace, MPItrace, OMPItrace, JIS

- Translators



Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002

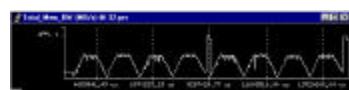
## Paraver: Performance Data browser

Raw data

tunable

Performance index :  $s(t)$  (piecewise constant)

Identifier of function  
Hardware counts  
Miss ratios  
Performance (IPC, Mflops,...)  
Routine duration  
...



Seeing is believing



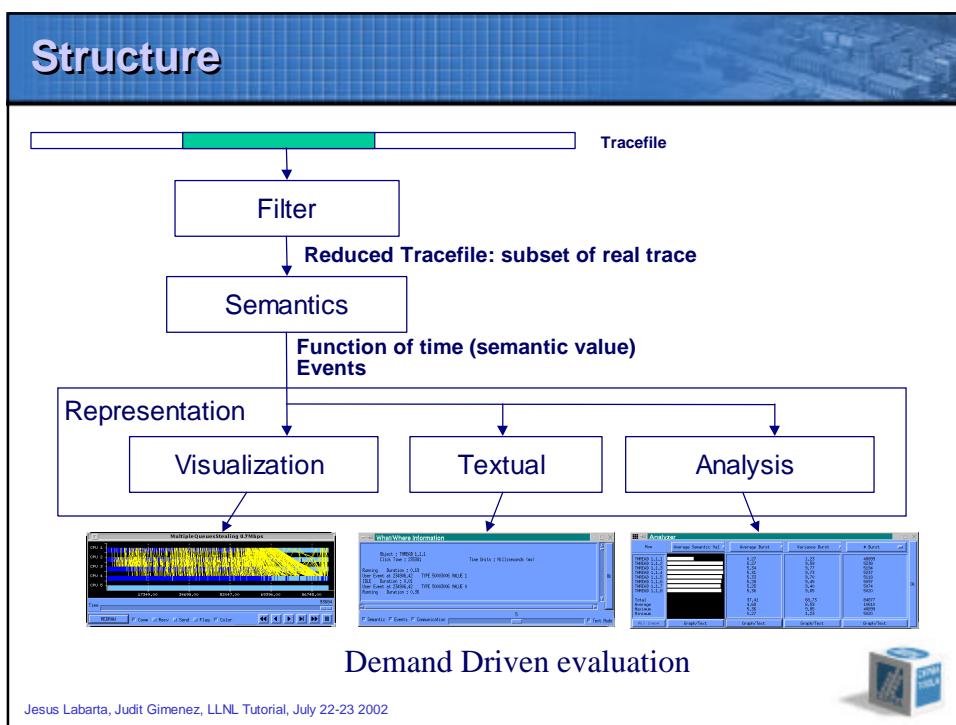
measuring is better

Statistics

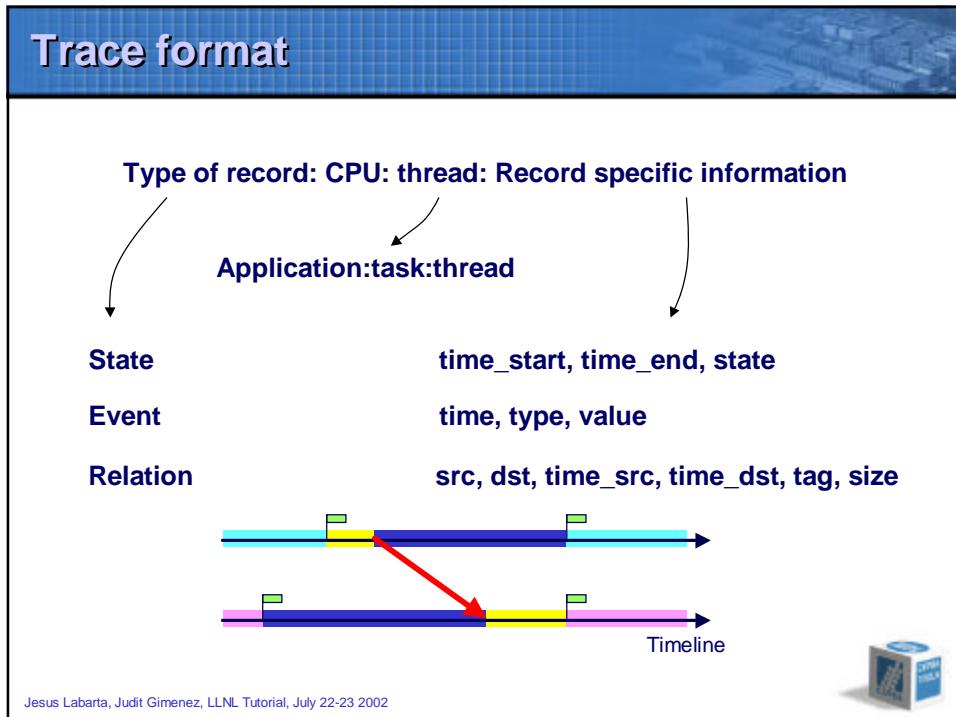
Average miss ratio per routine  
Histogram of routine duration  
Number of messages  
...

Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002

## Structure



## Trace format



## Trace records (.prv)

```
#Paraver (22/02/02 at 11:17):45221692:4:1:4(1:1,1:2,1:3,1:4)
2:1:1:1:1:0:40000001:1
2:1:1:1:1:0:50000001:1
1:1:1:1:1:0:3596:15
1:2:1:2:1:0:141:2
1:3:1:3:1:0:1413:2
1:4:1:4:1:0:4503:2
2:2:1:2:1:141:40000001:1
1:2:1:2:1:141:0:1
...
1:2:1:2:1:83562:84123:10
3:2:1:2:1:83562:84123:1:1:1:91028:111409:11520:2000
2:2:1:2:1:84123:50000022:0
1:2:1:2:1:84123:84132:1
2:2:1:2:1:84132:50000022:1
1:2:1:2:1:84132:84309:10
3:2:1:2:1:84132:84309:1:1:1:91019:111401:11520:3000
2:2:1:2:1:84309:50000022:0
1:2:1:2:1:84309:84314:1
2:2:1:2:1:84314:50000022:1
1:2:1:2:1:84314:84716:10
3:2:1:2:1:84314:84716:4:1:4:1:109427:113206:11520:4000
2:2:1:2:1:84716:50000022:0
```

Application:process:thread

Type and value

State

Size and tag

Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002



## Symbolic information (.pcf)

STATES	EVENT_TYPE
0 Idle	1 50000035 AllReduce (MPI)
1 Running	VALUES
2 Not created	0 End
3 Waiting a message	100 MPI_MAX
4 Blocked	101 MPI_MIN
5 Thd. Syncrh.	102 MPI_SUM
	103 MPI_PROD
DEFAULT_OPTIONS	104 MPI_LAND
LEVEL THREAD	105 MPI_BAND
UNITS MICROSEC	106 MPI_LOR
LOOK_BACK 100	107 MPI_BOR
SPEED 1	108 MPI_LXOR
FLAG_ICONS ENABLED	109 MPI_BXOR
DEFAULT_SEMANTIC	111 MPI_MAXLOC
THREAD_FUNC State As Is	110 MPI_MINLOC
EVENT_TYPE	
1 50000002 Buffered Send (MPI)	
1 50000003 Synchronous Send (MPI)	
1 50000004 Barrier (MPI)	
1 50000005 Broadcast (MPI)	
1 50000018 Send (MPI)	
1 50000019 Receive (MPI)	
VALUES	
1 Begin	
0 End	

Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002



## Visualization



### ■ Encoding

- Discrete colors / Function / Gradient color

### ■ Type of window

- Appl / Task / thread / Cpu...
- Object selection

### ■ Navigation



### ■ Zoom/Undo/Clone

### ■ Time measurement

- Within/ Between windows

### ■ Multiple traces and windows

- synchronize

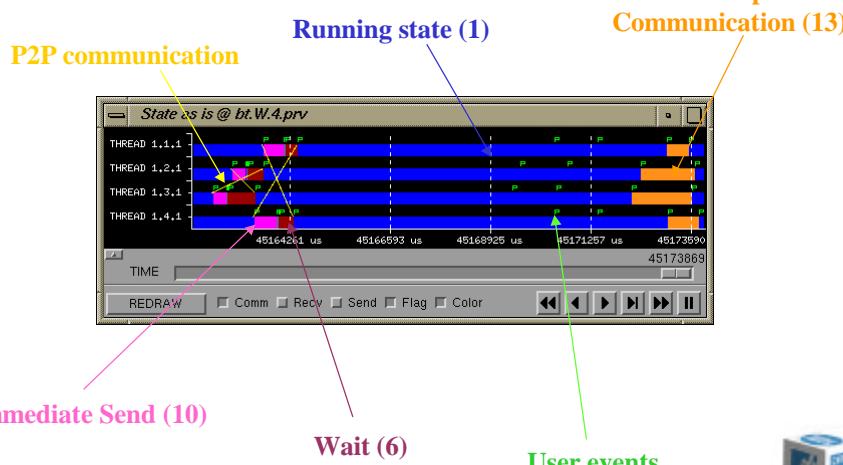
### ■ Y - scale



Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002

## Visualization: MPI

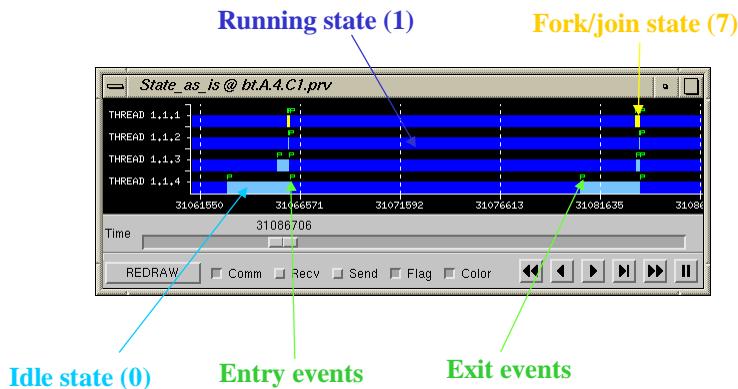
### ■ State display for MPI programs



Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002

## Visualization: OpenMP

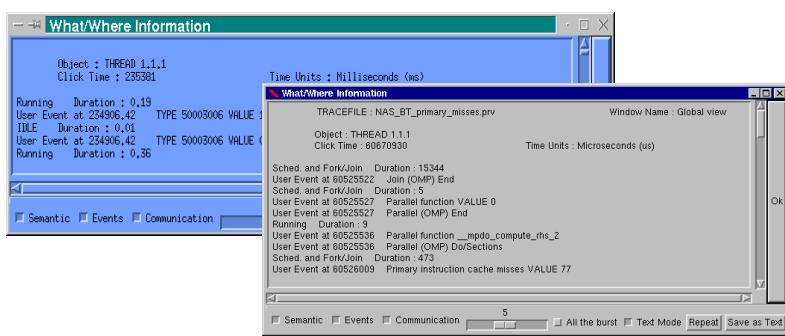
### ■ State display for OpenMP programs



Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002

## Textual

- Textual detail of area around point within window
- Semantic value and duration / flag / communication
- Numeric / translated text (.pcf file)

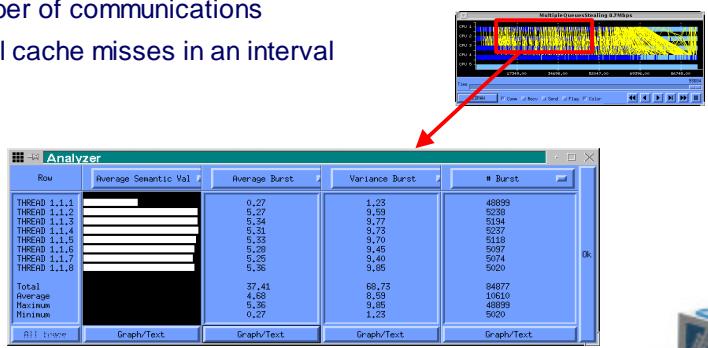


Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002

## Analysis modules: 1D

### ■ Example measures

- Average processor utilization
- Average duration/variance of specific function (if within range)
- Number of calls to specific function
- Number of communications
- Total cache misses in an interval
- ...



Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002

## Analysis modules: 2D

### ■ Single flexible quantitative analysis mechanism

### ■ Specified by:

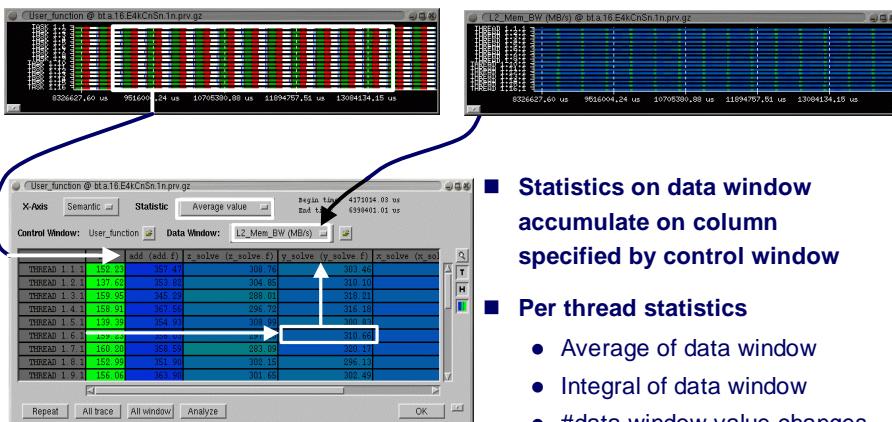
- Which statistic to compute
- On which performance index ? Data window
- How to present the data ? Control window

### ■ Several uses

- Standard profiling
- Hardware counters profile
- Function duration distribution
- Correlation between two performance indices

Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002

## Analysis modules: 2D



Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002



## Complex?

- **Window configuration files: Capture views**
  - Expert knowledge on how to compute performance index
  - Knowledge on “reasonable” values (scales)
  - Specific time and scales to expose behavior
- **What is useful for**
  - Performance analysis by non expert
  - Training
  - Checkpointing of studies
  - Cooperative work
  - Bug reporting

Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002



## Configuration files: Directory tree

### ■ Configuration files provided with OMPItrace

- General
  - ✓ State as is, user functions, user function distribution
- OpenMP
  - ✓ Parallel functions, parallel function distribution,
- MPI
  - ✓ MPI call profile, MPI call distribution, message size, send BW, ...
- Counters
  - ✓ Program
    - Memory ops mix, Memory access direction, ...
  - ✓ Architecture
    - L2 miss ratio, ...
  - ✓ Performance
    - IPC, cycles per ms, MIPS, Memory BW, processor BW, ...

Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002



## Semantic value

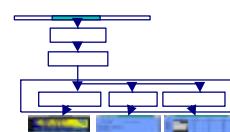
### ■ $S = f(t)$

- **Piecewise constant** function of time represented by one window
- Depends on the filtered subtrace: subset of records of the trace left through by the filter. Each window may see a different subtrace.
- The semantic value at time  $t$  may depend on records with time stamps potentially very far apart from  $t$ .

### ■ Let

- $t_i$ : ith instant of change
- $S_i$ : value taken by function at time  $t_i$

$$S(t) = S_i, i \in [t_i, t_{i+1})$$



Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002

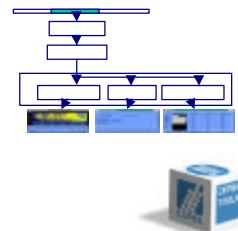
## Filter module

### ■ What : restrict records that pass to the semantic module

- Events
  - ✓ by type
  - ✓ by value
- Communications
  - ✓ by tag
  - ✓ by size
  - ✓ by source / destination
  - ✓ logical / physical

### ■ What for

- Reduce amount of information to display
- Feed properly the semantic module



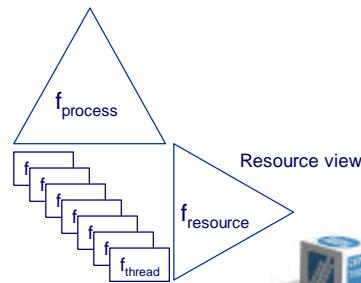
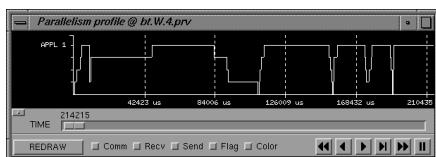
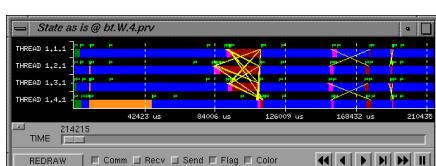
Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002

## Semantic module

### ■ Angle:

- Process model
  - ✓ Thread, task, application, workload
- Resource model
  - ✓ CPU, node, system

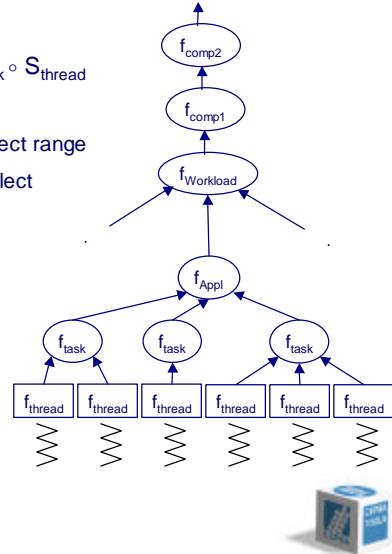
Process view



Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002

## Semantic module: process model view

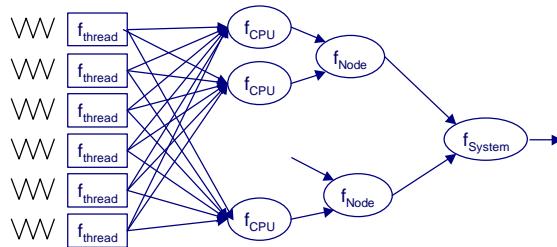
- Semantic value:  $S(t)$
- $S = f_{comp2} \circ f_{comp1} \circ f_{Workload} \circ f_{Application} \circ f_{task} \circ S_{thread}$
- Semantic functions
  - ✓  $f_{comp2}, f_{comp1}$ : sign, mod, div, in range, select range
  - ✓  $f_{Application}, f_{Workload}$ : add, average, max, select
  - ✓  $f_{task}$ : add, average, max, select
  - ✓  $S_{thread}$ : in state, useful, given state,  
last event value,  
next event value,  
average next event value  
interval between events, ...



Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002

## Semantic module: resource view

- $$Sf_{resource} = f_{comp2} \circ f_{comp1} \circ f_{System} \circ f_{Node} \circ f_{CPU} \circ S_{thread}$$
- Semantic functions
    - ✓  $f_{System}$ : add, average, max, select
    - ✓  $f_{Node}$ : add, average, max, select
    - ✓  $f_{CPU}$ : active thread, select
    - ✓  $S_{thread}$ : in state, useful, given state, next event value, thread\_id

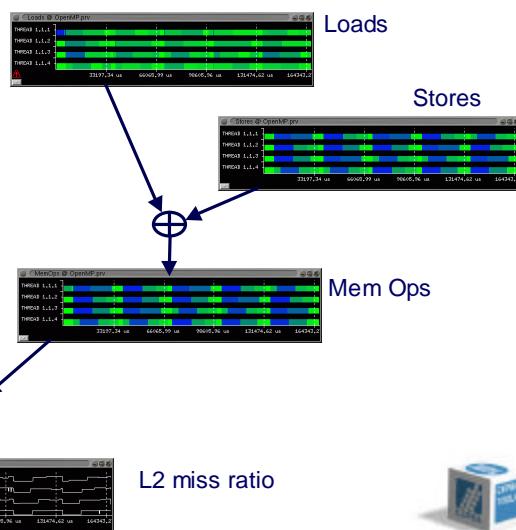


Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002

## Semantic module

### ■ Derived windows

- Point wise operation
  - ✓  $S = a * S^a * \text{op} > \beta * S^b$
  - ✓  $\text{op} : +, -, *, /, \dots$

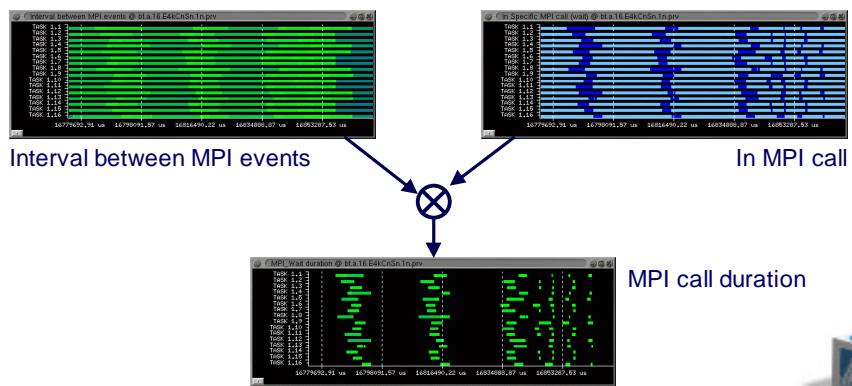


Jesus Labarta, Judit Gimenez, LLNL

## Semantic module

### ■ Derived windows

- Point wise operation
  - ✓  $S = a * S^a * \text{op} > \beta * S^b$
  - ✓  $\text{op} : +, -, *, /, \dots$



Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002

## OMPItrace: IBM Instrumentation

### ■ Tracing command

ompitrace [options] <normal command>

### ■ Probes

- Timing

- ✓ Clock switch if available
- ✓ System clock - read\_real\_time()

- Hardware counters

- ✓ Vendor specific: PMAPI

### ■ Insertion of probes

- Dynamic

- ✓ DPCL (<http://oss.software.ibm.com/developerworksopensource/dpcl>)

- Static:

- ✓ user explicit calls to OMPItrace API

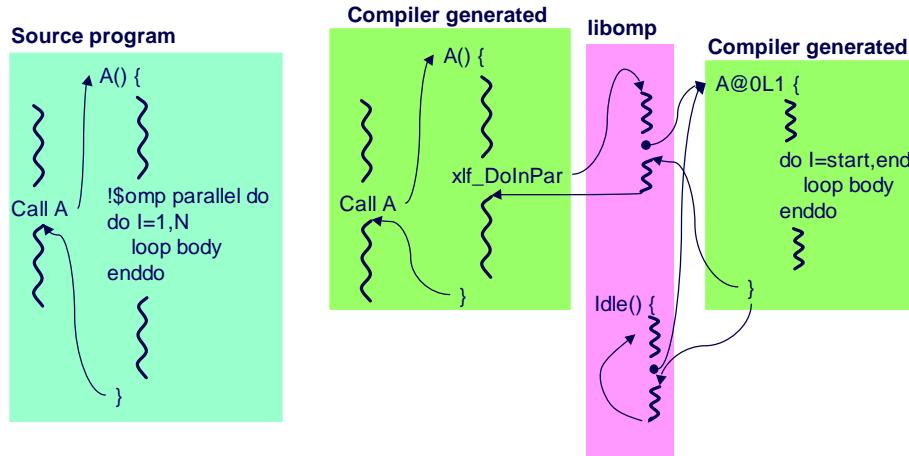
Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002



## OpenMP compilation and Run Time

### Source program

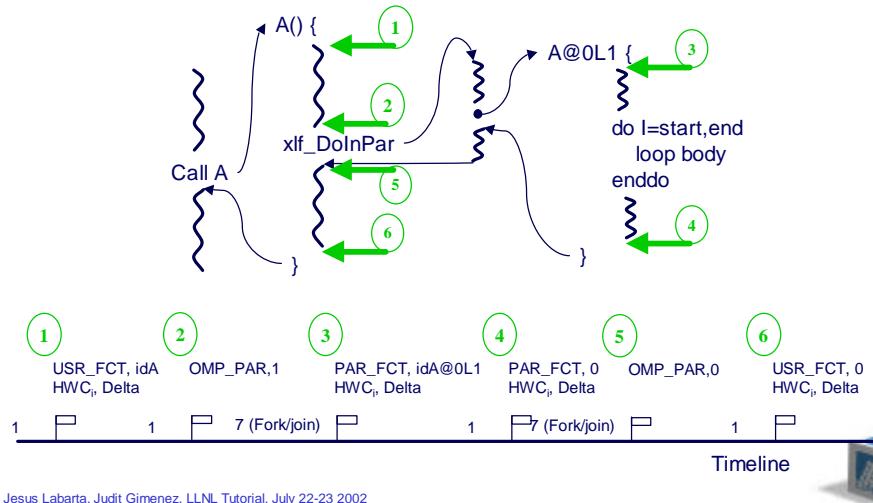
```
A() {  
    !$omp parallel do  
    do i=1,N  
    loop body  
    enddo  
}
```



Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002

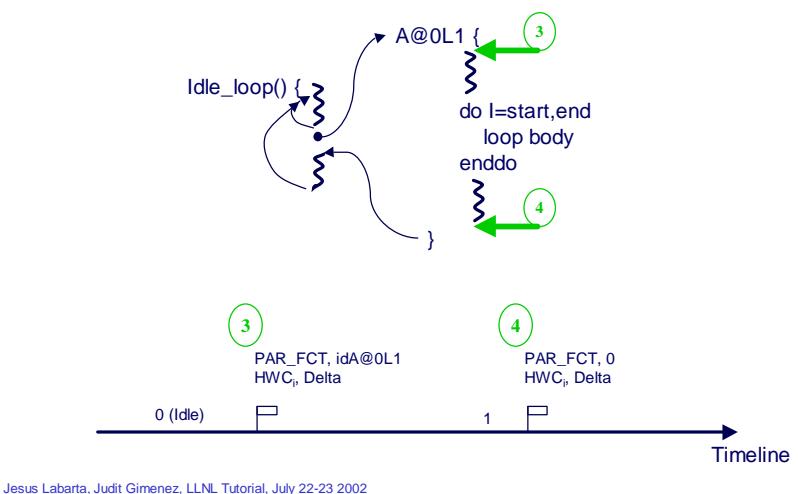
## OpenMP instrumentation points

### Main thread



## OpenMP instrumentation points

### Slave threads

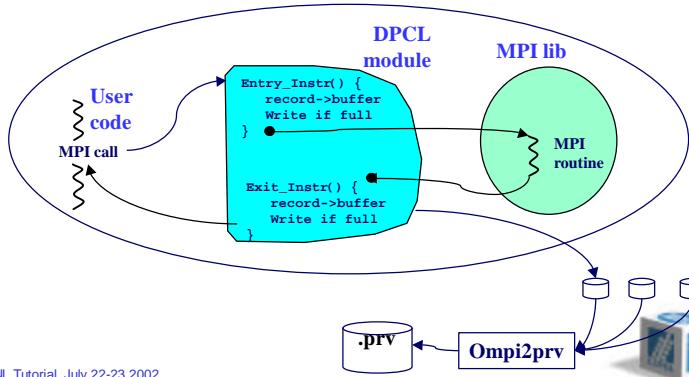


## MPI instrumentation

### ■ MPI call events

- identifier of the call on entry, 0 on exit
- Hardware counts on entry and exit if activated (-counters:mpi)

### ■ Probe insertion through DPCL



Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002

## OMPItrace API

### ■ emit events

- `ompitrace_event(int type, int value)`
- `ompitrace_eventandcounters(int type, int value)`
- `ompitrace_counters()`

### ■ stop/resume tracing

- `ompitrace_shutdown(), ompitrace_resume()`

### ■ Link with

- `-L$OMPITRACE_HOME/lib -lompitrace`

Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002



## OMPItrace: Overhead

### ■ Application elapsed time w/o instrumentation

- Similar → user happy
- Very different → the application has a problem
- Very different → still very useful
  - ✓ I.e.: Hardware counts

### ■ Learn how to live with it

- Don't relax, try to extract as much information as possible

### ■ Overhead of tracing in IBM

- No hardware counters: 2.7  $\mu$ s
- Hardware counters: 7.3  $\mu$ s

Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002



## OMPItrace: Overhead

### ■ Application elapsed time w/o instrumentation

- Sweep3d
- 8 threads

		No tracing	Tracing without hardware counters	Tracing with hardware counters
50^3	Diag	6.70	7.76	8.74
	Kjmi	3.21	3.30	3.24
125^3	Diag	76.82	79.96	81.97
	kjmi	37.57	37.58	37.94

Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002



## ute2paraver

### ■ UTE

- uses AIXtrace facility to obtain information of
  - ✓ MPI implementation
  - ✓ scheduling policies
  - ✓ resources mapping
- based on DPCL, but requires to link with libute.a

### ■ ute2paraver

- translate UTE traces to Paraver format
- similar but complementary view of OMPItrace

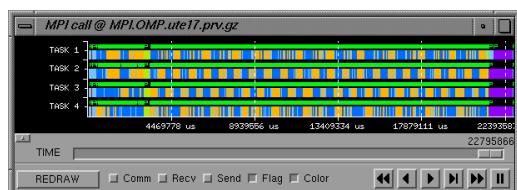
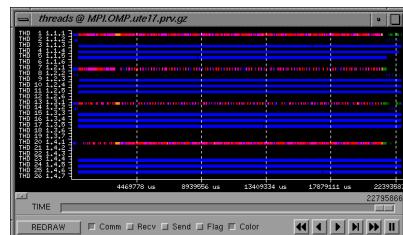
Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002



## ute2paraver views

### ■ Thread views

- Similar to OMPItrace
- Information on internal MPI threads



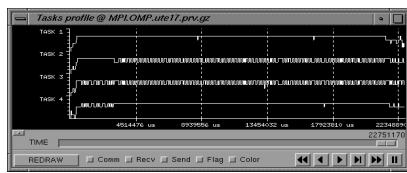
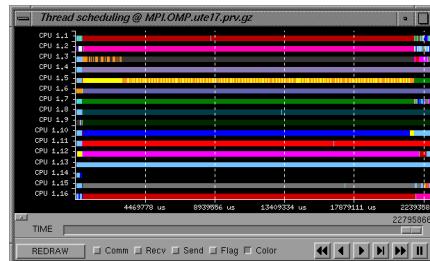
Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002



## Ute2paraver views

### ■ Resource view

- Information not provided by OMPtrace



Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002



## Conclusion

### ■ OMPtrace

- MPI + OpenMP tracing
- of unmodified production binaries

### ■ Ute2paraver

- PE Benchmark traces

### ■ Paraver

- Flexible trace browser
- Powerful quantitative analysis (2D)
- Configuration files
  - ✓ Support for fast views and analyses display

Jesus Labarta, Judit Gimenez, LLNL Tutorial, July 22-23 2002

